

ARGUMENTS

Rejection of Claims on Art Grounds in the 09/28/2004 Office Action, and Traversal Thereof

In the 09/28/2004 Office Action, claims 1-36 were rejected on prior art grounds, under 35 U.S.C. 102(a) and 35 U.S.C 103(a).

Claims 1-36 are pending. Claims 1-36 stand rejected for being non-statutory under 35 USC 101. Claims 1-5, 7-17 and 19-29 stand rejected under 35 U.S.C. 102(a, b) as being anticipated by **IBM** product Enterprise Javabeans Development Using VisualAge for JAVA, published June 1999. Claims 6 and 18 stand rejected under 35 U.S.C. 103(a) as being unpatentable over IBM as applied to claim 1-5, 7-9 and 10-15 above, and further in view of the design decision of adding a tab specifically for business method. Claims 6, 16 and 18 stand rejected by IBM in view of a design decision of adding a tab for business methods.

Claims 1, 10 and 19 are currently amended as suggested by the Examiner to overcome the rejection for being non-statutory under 35 USC 101. Claims 1 and 10 have been further amended to point out that the graphical representations of the code are diagrammatic as opposed to symbolic or iconic as disclosed by the prior art references. Claim 19 has been amended to include the limitation of claim 24. Claim 24 has been canceled and its dependent claims 25 and 27 have been amended to depend upon independent claim 19.

The present invention is a method and system for creating project plans from a workflow. More particularly, the invention is a method and system for integrating workflows and project plans to keep them “in step” with each other.

The Present Invention Is Novel Over IBM product Enterprise Javabeans Development
Using VisualAge for JAVA

As stated in MPEP §2131, a claim is anticipated under §102 only if each and every element as set forth in the claim, in as complete of detail, is found in a single prior art reference. The claimed invention, according to independent claim 1, as currently amended, calls for a step of:

displaying a diagrammatic graphical representation of the code that
includes a separately delineated display area for each of the plurality of
types.

As such, for the cited reference to be anticipatory of claim 1, the reference must describe this step. In other words, the reference, to teach in as much detail as is claimed by the present invention, must disclose a step of :

displaying a diagrammatic graphical representation of the code that
includes a separately delineated display area for each of the plurality of
types.

The Examiner holds the position that the tabs of Figure 10 on page 35 of the IBM reference and the tabs of Figure 17 on page 44 of the IBM reference both display a graphical representation of the code. However, the graphical representation of the code in the present invention is different than the graphical representation of the code as shown in Figures 10 and 17 of the IBM reference. The difference is that the graphical representation of the code as it pertains to the present invention is **diagrammatic**, whereas the graphical representation according to the IBM reference is symbolic or iconic. As currently amended, claim 1 requires a step of displaying a **diagrammatic**

graphical representation of the code. Examples of **diagrammatic** graphical representations of the code are shown in Figures 12-19 of the present application. The cited references do not disclose **diagrammatic** graphical representation of the code. Therefore, the cited references cannot anticipate presently amended independent claim 1 or its dependents.

The claimed invention, according to independent claim 10 as currently amended calls for a step of:

displaying a diagrammatic graphical representation of the method with the symbol indicating the type of the method.

As pointed out in the argument pertaining to claim 1, the cited references do not disclose displaying a **diagrammatic** graphical representation of the code. Therefore, the cited references cannot anticipate presently amended independent claim 10 or its dependents.

Claim 19 of the present application has been amended to clarify that the present invention includes a step of:

storing deployment information associated with the distributed computing component in a comment of the source code for the distributed computing component.

The cited IBM reference does not disclose the above underlined step in that the deployment descriptor file produced by the present invention is different than the deployment descriptor file produced under IBM's traditional approach. The difference is that the present invention stores properties in a comment code corresponding to a distributed computing component such as an Enterprise Java Bean (EJB). By storing a

group of EJB properties in a comment of code corresponding to the EJB, the software development tool may later retrieve the group of EJB properties as deployment information for the respective EJB. Thus, by storing deployment information for the respective EJB within a comment of code corresponding to the EJB, the software development tool allows one programmer to develop one EJB while a second programmer independently develops another EJB. Later, using the software development tool, the deployment information for the respective one EJB may be combined with the deployment information for the respective other EJB to generate a deployment descriptor file that jointly describes both EJBs for deployment to an EJB Application Server. In a traditional approach to development and deployment of EJBs, a respective EJB does not contain its own deployment information so all programmers must coordinate their development to produce EJBs using a single deployment descriptor file. Typically, just a single programmer at a time may access the single deployment descriptor file produced under the traditional approach such as disclosed in the IBM reference. Since the cited references do not disclose the underlined step in claim 19 that makes it possible for one programmer to develop one EJB while a second programmer independently develops another EJB, the cited references cannot be properly held to anticipate currently amended claim 19 and its dependents.

The Present Invention Is Not Obvious Over IBM

The cited publication to IBM broadly discloses steps involving CASE Tools. The methods and systems disclosed in the present application also disclose steps involving CASE Tools. However, neither the cited references nor the other references cited but not applied disclose, suggest or teach the above underlined steps. Therefore, the cited

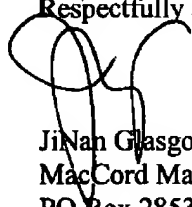
references, alone or in combination fail to contain any teaching or suggestion of the above underlined matter as it pertains to the present independent claims.

CONCLUSION

In view of the foregoing, claims 1-23 and 25-36 constituting the claims pending in the application, are submitted to be fully patentable and in allowable condition to address and overcome the rejections.

If any issues remain outstanding, incident to the allowance of the application, Examiner Ingberg is respectfully requested to contact the undersigned attorney at (919)-664-8222 or via email at jingang@trianglepatents.com to discuss the resolution of such issues, in order that prosecution of the application may be concluded favorably to the applicant.

Respectfully submitted,



Jinan Glasgow #42585
MacCord Mason, PLLC
PO Box 28539
Raleigh, NC 27611-8539
919-664-8222
919-664-8625 (fax)